engineering proceedings

MDPI

Proceeding Paper

Blender-Based Simulation and Evaluation Framework for GNSS-LiDAR Sensor Fusion †

Adam Kalisz *, Muhammad Khalil, Iñigo Cortés, Santiago Urquijo, Katrin Dietmayer, Matthias Overbeck, Christoph Miksovsky and Alexander Rügamer

Satellite Based Positioning Systems Department, Fraunhofer Institute for Integrated Circuits IIS, 90411 Nuremberg, Germany; muhammad.hamza.khalil@iis.fraunhofer.de (M.K.); inigo.cortes@iis.fraunhofer.de (I.C.); santiago.urquijo@iis.fraunhofer.de (S.U.); katrin.dietmayer@iis.fraunhofer.de (K.D.); matthias.overbeck@iis.fraunhofer.de (M.O.); christoph.miksovsky@iis.fraunhofer.de (C.M.); alexander.ruegamer@iis.fraunhofer.de (A.R.)

* Correspondence: adam.kalisz@iis.fraunhofer.de

† Presented at the European Navigation Conference 2025 (ENC 2025), Wrocław, Poland, 21–23 May 2025.

Abstract

The fusion of Global Navigation Satellite System (GNSS) and Light Detection and Ranging (LiDAR) sensors has emerged as a critical research area for high-precision navigation and mapping applications. While GNSS provides absolute positioning, it is susceptible to multipath errors, signal occlusions, and atmospheric disturbances. LiDAR, on the other hand, offers high-resolution environmental perception but lacks absolute localization and is sensitive to sensor noise and drift over time. To address these limitations, robust sensor fusion architectures are necessary to improve positioning accuracy, reliability, and robustness in diverse environments. This research focuses on the systematic modeling of GNSS and LiDAR errors to enhance sensor fusion performance. A key aspect of this work is the design of fusion architectures that optimize trade-offs between accuracy, environmental-dependency, and robustness to sensor failures. To this end, this research investigates trajectory alignment, geometric similarity, and sensor signal dropouts. Various fusion strategies, including tightly coupled and loosely coupled approaches, are explored to evaluate their effectiveness under different operational conditions. Simulation-based evaluation is a core component of this study, enabling controlled analysis of sensor errors, fusion methodologies, and performance metrics. A custom Blender-based simulation framework has been developed to facilitate reproducible experiments and allow for the benchmarking of different fusion strategies. By systematically analyzing fusion performance in terms of accuracy, consistency, and computational cost, this work aims to provide valuable insights into the optimal integration of GNSS and LiDAR for real-world applications. The simulation framework generates a reusable output format in order to demonstrate the flexibility of this methodology by running a selected fusion approach on real data (Sim2Real). The proposed framework and findings contribute to the research community by providing tools and methodologies for evaluating sensor fusion strategies, fostering advancements in precise and resilient localization solutions for autonomous systems, robotics, and geospatial applications in challenging environments.

Keywords: GNSS; LiDAR; sensor fusion; blender; simulation

Check for updates

Academic Editor: Tomasz Hadaś

Published: 14 February 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Autonomous navigation systems must operate reliably in diverse environments—from open highways to densely built urban centers and intricate waterways. The fusion of data from the Global Navigation Satellite System (GNSS) with Light Detection and Ranging (LiDAR) sensor measurements promises significant improvements in localization and mapping. However, achieving robust integration requires addressing concrete challenges related to sensor error modeling, particularly in the context of Simultaneous Localization and Mapping (SLAM).

This research is motivated by the sensor data recording during the Smart Port Shuttle (SPS) [1] project from 2023. It focused on developing and evaluating logistic concepts for semi-autonomous vessels in inland navigation. The safety and reliability of these vessels require a highly precise positioning system. Consequently, a multi-sensor system consisting of GNSS receivers [2], LiDARs, camera, radar, and IMU was developed. The system was integrated into the Robot Operating System (ROS) in order to allow the recording, processing, and evaluation of the complete sensor setup in real time as can be seen in Figure 1.

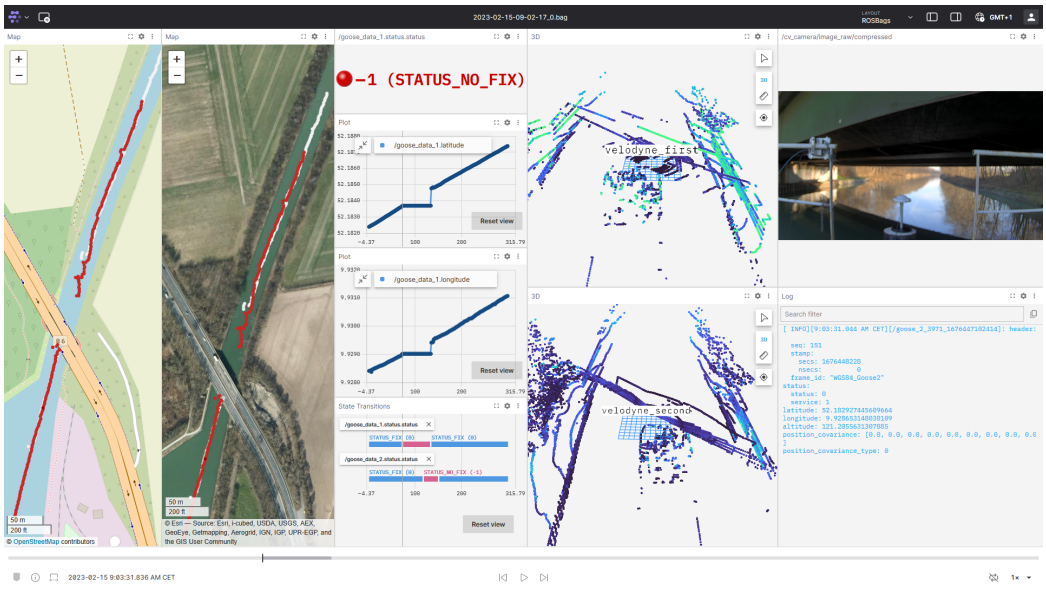


Figure 1. ROS bag recording of a water canal scene from the SPS measurement campaigns. Red and white points represent GNSS measurements from two receivers. The image illustrates a GNSS-denied environment as soon as a ship approaches the bridge (see map views on the left); both GNSS receivers loose satellite signal, resulting in missing position data.

Many autonomous systems rely on satellite-based positioning for navigation [3]. Standalone GNSS receivers generally provide Position, Velocity, and Time (PVT) information to the user. However, GNSS-based positioning can be degraded in urban canyons, tunnels, indoor, and forests, resulting in wrong and/or no position information [4]. The same effect was observed in the SPS project. When the ship approaches the bridge, a GNSS receiver was completely occluded, resulting in a missing position estimate. Secondly, single-antenna GNSS receivers cannot provide orientation information or a high-definition geometric map of the environment. LiDAR sensors can provide orientation and mapping information but highly depend on the reflectometric and geometric characteristics of the environment. To this end, this paper investigates the fusion of GNSS and LiDAR sensors by discussing open challenges, outlining the contributions by introducing a blender-based framework and presenting the evaluation of both simulated and real data with it.

https://doi.org/10.3390/engproc2026126021

1.1. LiDAR SLAM Challenges

Dynamic Data Association and Feature Matching

In dynamic and cluttered environments, the robust extraction and matching of geometric features between consecutive scans remains a significant challenge. Algorithms frequently suffer from failures in data association when faced with moving objects, occlusions, or repetitive patterns.

Loop Closure Detection and Global Consistency

Detecting loop closures reliably is critical for correcting drift in LiDAR SLAM. However, in large-scale or repetitive environments—especially urban canyons—existing techniques often fail to identify loop closures, leading to accumulated drift and global inconsistency.

Scalability and Real-Time Performance

As the mapping area increases, the computational load and memory footprint of LiDAR SLAM systems become critical issues. Maintaining real-time performance while processing large amounts of data in complex environments is still an unresolved challenge.

1.2. GNSS Challenges

Multipath and Satellite Occlusion

GNSS measurements can be severely degraded by multipath, non-line-of-sight reflections and satellite occlusions [5], especially in urban environments or in scenarios like bridges over waterways. These issues result in large errors and inconsistent pseudorange measurements, challenging fusion systems that rely on GNSS data.

Jamming and Spoofing Attacks

GNSS signals are particularly vulnerable to jamming and spoofing [6] due to their low received power and open-access transmission, making them susceptible to interference that can either overwhelm the signal or mislead receivers with counterfeit data. These attacks introduce significant challenges in maintaining reliable positioning, as jamming can cause loss of signal lock and spoofing can result in erroneous, manipulated positions.

Measurement Noise and Uncertainty Propagation

Accurate modeling of the sensor noise—particularly the non-Gaussian and time-varying characteristics of GNSS—is a persistent problem. This inaccuracy leads to inconsistent uncertainty estimates, which undermine the reliability of the fusion system's covariance predictions.

1.3. Fusion Challenges

Coordinate Frame Misalignment

A common challenge is the misalignment between the LiDAR SLAM trajectory and the GNSS coordinate frame. This misalignment can be caused by sensor calibration errors, drift in the LiDAR system, or differences in the assumed reference frames.

Robustness to Degraded Sensor Conditions

Fusing data from heterogeneous sensors requires the fusion system to be robust against sensor degradation. When one sensor (e.g., GNSS) is unreliable due to poor satellite geometry or multipath effects, the system must adaptively re-weight the inputs and maintain overall estimation accuracy.

Sim2Real Gap and Model Adaptation

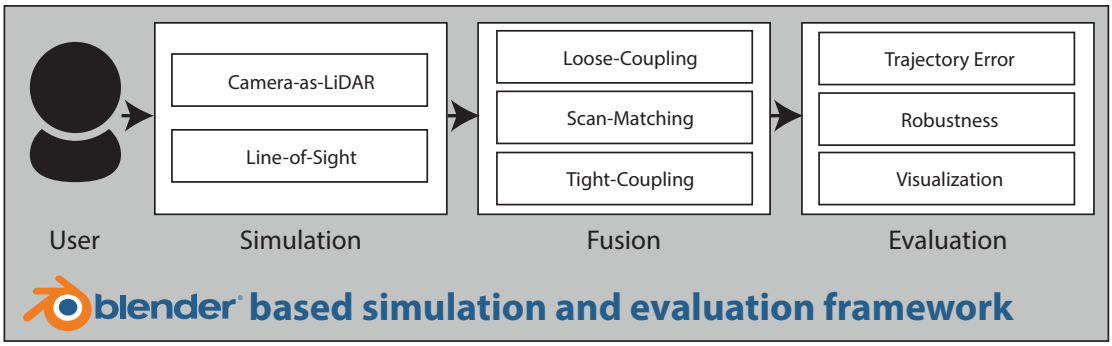
One of the most critical challenges is the disparity between simulated and actual real-world sensor behavior. Accurately modeling environmental effects and sensor characteristics in simulation is difficult. The research community has emphasized the need to continuously calibrate and adapt these models based on real-world feedback.

https://doi.org/10.3390/engproc2026126021

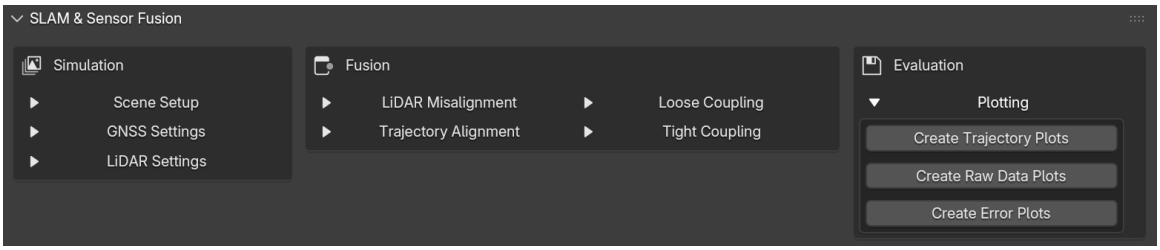
2. Materials and Methods

Investigating the fusion of GNSS and LiDAR sensors is crucial for advancing SLAM and autonomous navigation technologies. By addressing specific challenges—such as satellite occlusion when tracking ships under bridges and repetitive structures in LiDAR point clouds—researchers can design more reliable and accurate fusion algorithms. Detailed sensor error modeling, combined with high-fidelity simulation validation using tools like Blender, creates a direct pathway from theoretical research to practical, real-world application, paving the way for the next generation of resilient autonomous systems.

Based on the scientific challenges presented, our work makes the following contributions, briefly visualized in Figure 2.



(**a**)



(**b**)

Figure 2. Proposed method: (**a**) Flowchart depicting the user-centric process from simulation and fusion design to evaluation. (**b**) Graphical user interface of the framework based on the flowchart above. The interface is organized by task, with each section providing detailed parameters such as Keplerian elements for satellite orbit generation and sensor noise specifications.

2.1. Context-Specific Sensor Error Modeling

GNSS: We develop detailed models for GNSS errors in scenarios where satellite visibility is intermittently blocked—such as when ships navigate under bridges depicted in Figure 3—capturing the effects of satellite occlusion and increased dilution of precision (DOP).

LiDAR: We establish quantitative error models for LiDAR measurements under challenging conditions (e.g., low-reflectivity surfaces like wet decks or uniform canal walls). Our approach uses the LiDAR-as-Camera method [7] and separates environmental information into different render passes shown in Figure 4a. Consequently, errors based on depth, semantics, surface normals, or speed vectors can be introduced by the modification of ideal data. Our approach enables precise characterization of environmental influences—such as noise and ambiguities—in the generated point cloud data shown in Figure 4b.

https://doi.org/10.3390/engproc2026126021

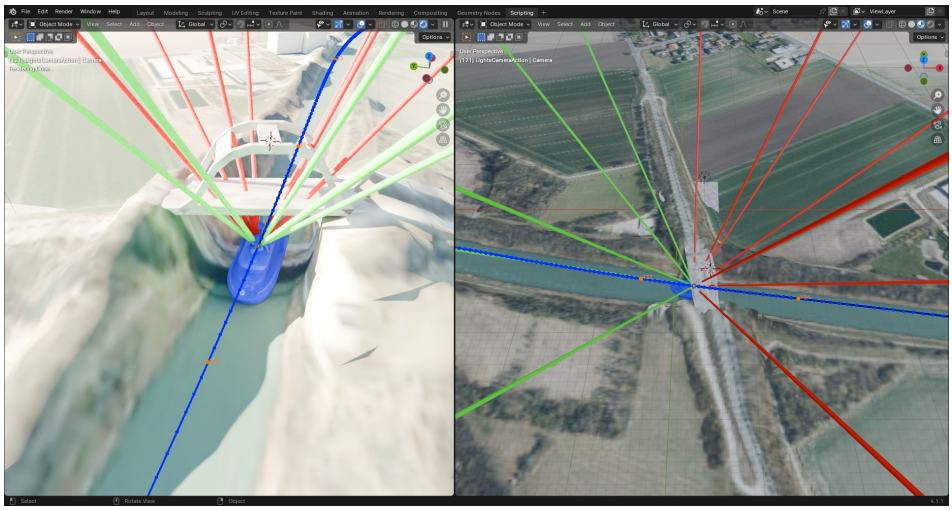
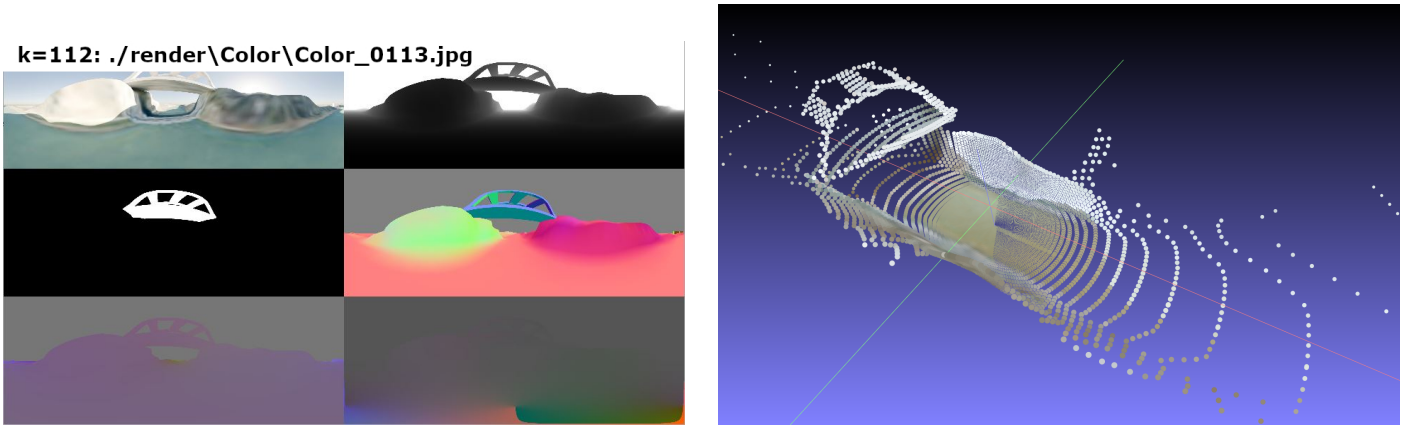


Figure 3. Synthetic water canal scenario inspired by the Smart Port Shuttle (SPS) measurement campaigns. An artist creates the scene and animates the pose of the sensor (depicted as a ship) in Blender. Our framework is automatically generating the ground truth, GNSS pseudoranges, skyplot, and Line-Of-Sight (LOS) information (green rays: LOS, red rays: Non-LOS) together with the standard vision data such as color, depth, and semantic information.



(**a**) (**b**)

Figure 4. LiDAR data generation: (**a**) Synthetic data rendered as camera data. From top left to bottom right: Color, depth, segmentation mask, normal, position, and vector render passes. (**b**) The final LiDAR data generated for each of the synthetic images. The point clouds are available as timestamped .ply files which are identical to the extracted real data from ROS bags.

2.2. Adaptive Fusion Framework for SLAM

We propose a dynamic sensor fusion algorithm that integrates GNSS and LiDAR data for the investigation of suitable fusion error metrics (e.g., GNSS dilution of precision levels and a combination of the number of iterations and root-mean-square-error from LiDAR odometry). Table 1 summarizes state-of-the-art coupling types for sensor fusion. Specifically, we use a loosely coupled approach where each sensor is processed individually. The absolute position estimate from GNSS is fused with the relative SE(3) Lie Group from small_gicp [8] in both Extended Kalman Filter (EKF) and Factor Graph (FG) formulations.

Table 1. Types of sensor coupling for GNSS and LiDAR fusion.

Coupling Type	Definition	Example for GNSS & LiDAR
Loosely Coupled	Fusion at the solution level; each sensor is processed independently to produce an estimate, which is then fused.	Compute a GNSS position solution independently and a LiDAR odometry (or SLAM) estimate independently, then fuse them (e.g., using an EKF in the `robot_localization` package) [9].
Tightly Coupled	Fusion at the raw measurement level; sensor measurements are integrated directly within a single estimation framework.	Directly integrate raw GNSS observations (e.g., code-phase) with LiDAR scan matching constraints in a unified factor graph optimization (e.g., using GTSAM or iSAM2) [10,11].
Hybrid Coupled	Combination of both loosely and tightly coupled approaches; some sensor data are fused at the raw level while others are fused as processed estimates.	Fuse LiDAR-based odometry tightly within a SLAM framework and incorporate GNSS position fixes in a loosely coupled manner to correct global drift, for example as initialization for the optimization algorithm.

2.3. High-Fidelity Simulation for sim2real Validation

We leverage Blender's advanced modeling, animation, and rendering capabilities to create diverse simulation environments that accurately reproduce scenarios like satellite occlusion under bridges. This enables the rigorous testing and validation of fusion algorithms by comparing synthetic data—where sensor error characteristics are precisely injected—to actual field measurements.

2.4. Empirical Analysis and Performance Benchmarking

Through extensive simulations and field experiments, we provide a comprehensive evaluation of the proposed sensor fusion framework. Our analysis quantifies improvements in localization accuracy and robustness in mixed environments, thereby demonstrating the practical benefits of our error modeling and fusion strategies.

These contributions collectively advance the state of the art in GNSS–LiDAR fusion for SLAM, paving the way for more resilient and accurate autonomous navigation systems in complex, real-world settings. The proposed user interface is shown in Figure 2b.

3. Results

Figures 5–7 summarize synthetic and real scenarios. The synthetic scenarios were generated using the proposed framework. Both datasets have the same file structure, i.e., timestamped .ply files for LiDAR scans and .npy files for GNSS measurements, whereas the real data were converted to these files from a ROS bag recording. In the case of real data, the GNSS constellation creation and PVT calculation can be skipped and the user continues with data analysis. Once GNSS and LiDAR SLAM results have been calculated, various plots—for example, the individual trajectories; skyplot; Dilution of Precision (DoP); Root Mean Square Error (RMSE); D-, A-, or E-optimality criteria [12]; and number of iterations or convergence flags—can be generated by clicking the respective button in the user interface. This allows researchers to investigate possible sensor error models based on different situations. These visualizations are not included here to maintain focus and readability.

Table 2 summarizes the SLAM and fusion performance briefly. In scenario 1, depicted in Figure 5, we have created a digital twin of a real inland waterway demonstrating good performance for LiDAR SLAM but poor GNSS performance when the receiver approaches

a bridge. The missing PVT fix during GNSS blockage can easily be bridged using the LiDAR sensor during the GNSS signal outage.

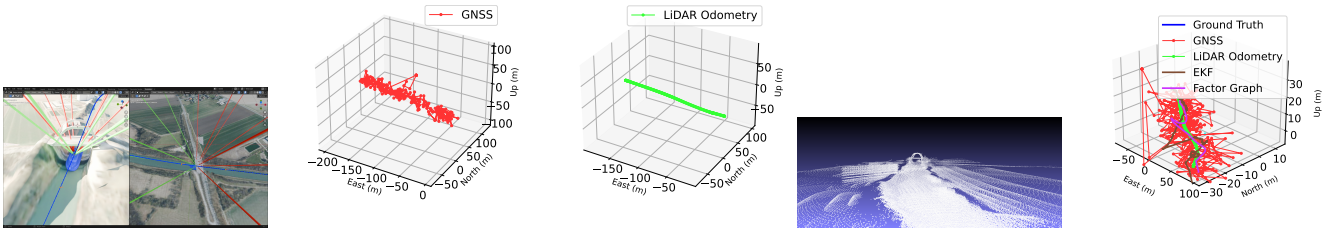
Table 2. Evaluation of different scenarios regarding the number of laser scans, average runtime of laser SLAM, and RMSE between the Extended Kalman Filter (EKF) or Factorgraph (FG) and Groundtruth.

Scenario	Number of Scans	Average Runtime of small_gicp per Scan (ms)	EKF RMSE (m)	FG RMSE (m)
1: Synthetic SPS	250	4.803	6.056	4.451
2: Real SPS	2973	22.133	85.965 *	26.407 *
3: Repetitive	250	2.412	1.638	1.509

* Real data is compared to iNAT-FSLG-01/T (RTK GNSS/INS) from iMAR Navigation GmbH, St. Ingbert, Germany.

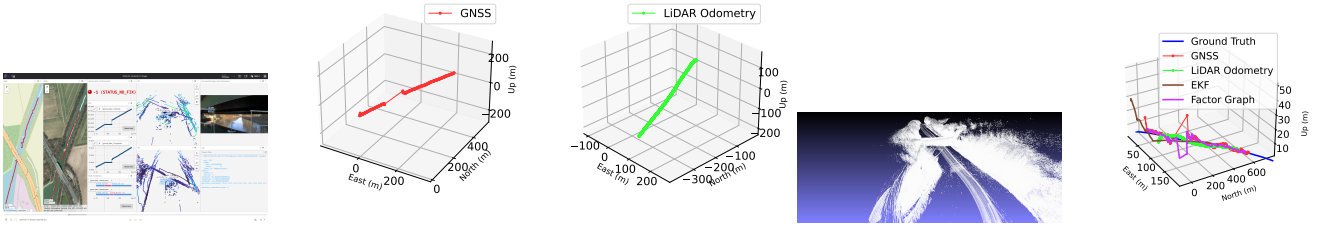
Scenario 2 shows one of the real measurement campaigns recorded in Hildesheim, Germany. Even though the laser scanner point clouds contain parts of the moving ship, dynamic water surfaces, and the shoreline contains intricate vegetation details, the SLAM algorithm works well in this environment. However, with an increase in measurements, the estimated trajectory (and thus resulting map) tends to drift downwards as the estimations accumulate rotational errors when the structure of the environment becomes narrower and lacks vertical structures for drift correction such as bridges. Due to the higher number of laser scans in the real-world dataset compared to the synthetic scenarios, the overall RMSE derived from the Absolute Trajectory Error (ATE) is increased as errors accumulate over more frames. On the contrary, the bridge is imposing challenges for GNSS resulting in no fix. Still, in this situation, the fusion of both sensors provides a continuous position estimate.

In scenario 3, we found that the LiDAR SLAM reports a convergence for all point clouds. However, as can be seen in Figure 7c, the estimated laser odometry is clearly wrong due to the repetitive environment. In such situations, the proposed framework allows us to design a suitable error metric for fusion methods and evaluate their performance on a wide range of different scenarios.



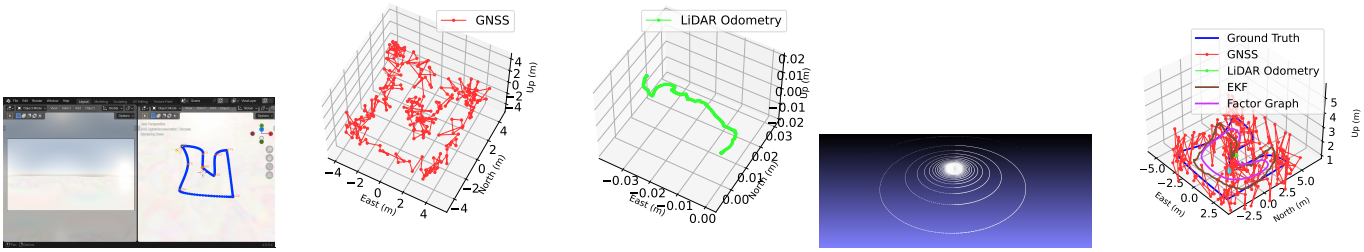
(**a**) (**b**) (**c**) (**d**) (**e**)

Figure 5. Evaluation of synthetic SPS scenario (scenario 1): (**a**) Scenario overview; (**b**) GNSS PVT trajectory; (**c**) LiDAR SLAM trajectory; (**d**) LiDAR SLAM point cloud; and (**e**) GNSS-LiDAR loose vs. tight coupling.



(**a**) (**b**) (**c**) (**d**) (**e**)

Figure 6. Evaluation of real SPS scenario (scenario 2): (**a**) Scenario overview; (**b**) GNSS PVT trajectory; (**c**) LiDAR SLAM trajectory; (**d**) LiDAR SLAM point cloud; and (**e**) GNSS-LiDAR loose vs. tight coupling.



(**a**) (**b**) (**c**) (**d**) (**e**)

Figure 7. Evaluation of synthetic scenario with repetitive structure, i.e., a perfectly planar ground (scenario 3): (**a**) Scenario overview; (**b**) GNSS PVT trajectory; (**c**) LiDAR SLAM trajectory; (**d**) LiDAR SLAM point cloud; and (**e**) GNSS-LiDAR loose vs. tight coupling.

4. Conclusions

This research proposed a flexible simulation and evaluation framework for sensor fusion which was created using the open source software Blender 4.5 LTS. It is a novel approach which enables users to intuitively manipulate all aspects of the simulation—from scenario design and sensor configuration to algorithmic tuning and performance evaluation—within a single, fully integrated software environment. Scenarios can either be imported from real data provided as ROS bags or photogrammetry and created synthetically by Blender artists where specific characteristics such as dynamic objects, repetitive environments, or satellite geometries need to be available. It is clearly aimed at improving our understanding of sensor modeling and data fusion design while still being easy to use for users with a basic understanding of Blender.

The framework is designed to be extensible for individualized research if necessary. Currently, the framework implements its core features such as GNSS constellation creation and PVT calculation, laser scan densification and registration, data fusion, and plotting for data analysis using only pure python with its popular libraries such as numpy and matplotlib in order to keep the dependencies to a minimum. For the estimation of laser odometry via SLAM, the aforementioned small_gicp package was used. In case researchers need to investigate other SLAM or data fusion algorithms, for example, the ones mentioned in the literature review, the framework can be extended via a respective python wrapper.

From a research perspective, we have investigated how GNSS pseudoranges are influenced via raycasting within (non-)line-of-sight environments and their impact on the least-squares calculation of the PVT. Full control over receiver and satellite location in addition to an adjustable environment enables statistical analysis of environmental dependency within GNSSs. Our approach is, however, limited to a simple geometric analysis of primary rays between satellites and receiver. This ignores tracing secondary rays using more sophisticated physical models such as Blender's probabilistic pathtracer called Cycles. Combined with radio frequency-based propagation, this could allow for multipath analysis. In future work, we consider extending the framework with more sophisticated physically based simulation models and allow users to add multi-frequency jammers and spoofer objects into the simulation.

Moreover, our approach is generating ideal laser data from render passes which comes with the advantage that, for each measurement, we store various additional data besides the usual range information. For example, users may assign specific IDs to 3D objects in the scene which can then be used to add semantic information to point clouds as depicted in Figure 4a. Semantic information is used to train object detection models and to increase the robustness of SLAM algorithms in dynamic scenes. Our approach can be used to evaluate semantic SLAM algorithms or fuse semantic point clouds with

GNSS measurements. In future work, we plan to provide a more thorough investigation of how single GNSS measurements and LiDAR point clouds correlate in order to augment GNSS with detailed information about the environment. One idea is to co-register LiDAR point clouds—first semantically segmented into classes like water surface, dock structures, vegetation, and vessels—with time-stamped GNSS fixes to quantify how environmental features drive GNSS error and uncertainty. This could yield a predictive map of "multipath hot spots" (e.g., highly reflective metallic pilings or concrete quay walls) that degrade GNSS performance and provide a real-time weight adjustment for each satellite fix in a fusion filter.

Author Contributions: Conceptualization, A.K. and M.K.; methodology, A.K.; software, A.K. and M.K.; data curation, A.K. and M.K.; writing—original draft preparation, A.K.; writing—review and editing, A.K., K.D. and A.R.; visualization, A.K.; supervision, I.C. and S.U.; project administration, K.D., M.O., C.M. and A.R.; funding acquisition, M.O., C.M. and A.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ICP	Iterative Closest Point
RMSE	Root Mean Square Error
SLAM	Simultaneous Localization And Mapping
SPS	Smart Port Shuttle

References

1. Dietmayer, K.; Schwenninger, J.; Gupta, H.; Saad, M.; Lipka, M.; Overbeck, M.; Felber, W. Smart Port Shuttle: Sensor-Based Navigation for Inland Waterway Transportation. *Eng. Proc.* **2023**, *54*, 50. [CrossRef]
2. Garzia, F.; Strobel, C.; Overbeck, M.; Kumari, N.; Joshi, S.; Förster, F.; Felber, W. A multi-frequency multi-constellation GNSS development platform with an open interface. In Proceedings of the 2016 European Navigation Conference (ENC), Helsinki, Finland, 30 May–2 June 2016; pp. 1–7. [CrossRef]
3. Wang, W.; Fernández-Gutiérrez, D.; Doornbusch, R.; Jordan, J.; Shan, T.; Leoni, P.; Hagemann, N.; Schiphorst, J.K.; Duarte, F.; Ratti, C.; et al. Roboat III: An autonomous surface vessel for urban transportation. *J. Field Robot.* **2023**, *40*, 1996–2009. [CrossRef]
4. Shen, W.; Yang, Z.; Yang, C.; Li, X. A LiDAR SLAM-Assisted Fusion Positioning Method for USVs. *Sensors* **2023**, *23*, 1558. [CrossRef] [PubMed]
5. Kaplan, E.; Hegarty, C.J. *Understanding GPS/GNSS: Principles and Applications*, 3rd ed.; Artech House, Inc.: Norwood, MA, USA, 2017.
6. Babic Rados, K.; Balić, M.; Begusic, D. Recent Advances on Jamming and Spoofing Detection in GNSS. *Sensors* **2024**, *24*, 4210. [CrossRef] [PubMed]
7. Tampuu, A.; Aidla, R.; van Gent, J.A.; Matiisen, T. LiDAR-as-Camera for End-to-End Driving. *Sensors* **2023**, *23*, 2845. [CrossRef] [PubMed]
8. Koide, K. small_gicp: Efficient and parallel algorithms for point cloud registration. *J. Open Source Softw.* **2024**, *9*, 6948. [CrossRef]
9. Moore, T.; Stouch, D. A Generalized Extended Kalman Filter Implementation for the Robot Operating System. In Proceedings of the 13th International Conference on Intelligent Autonomous Systems (IAS-13), Padova, Italy, 15–18 July 2014.
10. Dellaert, F.; Agrawal, V.; Roberts, R.; Cunningham, A.; Beall, C.; Ta, D.-N.; Jiang, F.; Lucacarlone; Nikai; Blanco-Claraco, J.L.; et al. borglab/gtsam. 2022. Available online: https://zenodo.org/records/8313530 (accessed on 14 February 2026). [CrossRef]

11. Kaess, M.; Johannsson, H.; Roberts, R.; Ila, V.; Leonard, J.; Dellaert, F. iSAM2: Incremental smoothing and mapping with fluid relinearization and incremental variable reordering. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 3281–3288. [CrossRef]
12. Pukelsheim, F. *Optimal Design of Experiments*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2006. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.